The invention relates to a method of producing electronic security modules and to the security modules produced in accordance with this method. Chip cards, which can be used for example as a means of payment or as signature cards, must be configured by means of specific prescribed methods in such a way that no misuse is possible.

Various entities are involved in the manufacture of such a chip card. Firstly, there is the chip manufacturer who thus produces the core part of the chip card. A ROM mask is then applied to the chip, said ROM mask being supplied by a different manufacturer. The ROM mask contains among other things the operating system which is required for operation of the chip card.

In the final process of manufacturing the chip card, said chip card must firstly be initialized and then personalized. During initialization, the conditions are met for loading all the personalization data into the

memory area of the chip. In the process, all the globally required data are transferred and the necessary file structures are created.

During the subsequent personalization, the individual data are introduced into the chip card. The cards are then supplied by the purchasers, for example credit distributors, to banks or directly to end customers.

During the personalization, it must be ensured that the data belonging thereto cannot be intercepted. The initialization and personalization operations are therefore treated as separate process steps and are also carried out at different locations.

It is an object of the invention to further improve an electronic security module, in particular a chip card, in terms of security technology.

In order to achieve this object, the invention proposes a method having the features specified in Claim 1. The invention also proposes a security module which can be manufactured in accordance with this method. Further developments of the invention form the subject matter of dependent claims.

Based on checking the signature relating to specific data with the aid of the public key of the purchaser of the chip card which is stored in the chip, it can be ensured whether the initialization data actually originate from the correct location.

Instead of the public key itself, it is also possible, as proposed in one embodiment of the invention, for a hash value derived from the public key to be introduced during chip manufacture.

The hash value is a test value which makes it possible to detect changes in the public key. Two different public keys in practice always have a different hash value. However, it is not possible to deduce from the hash value the key from which the hash value was derived. In this way, it becomes possible during the initialization to check whether the initialization data actually originate from the correct location, that is to say from the correct purchaser of the chip card. If a check reveals that the hash value and the public key do not match, the initialization is terminated.

The hash value derived from the public key has the advantage that it takes up less space than the public key itself.

In one embodiment of the invention, it may be provided that the hash value is generated by the purchaser of the chip card and the manufacturer of the chip and/or of the ROM mask is informed thereof.

It may be provided that the algorithm used to calculate the hash value or details regarding the known algorithm which was used is/are given to the manufacturer of the chip and/or of the ROM mask and is also stored in the memory of the chip.

It is also possible and within the scope of the invention that the hash value is generated by the manufacturer of the chip and/or of the ROM mask and is stored in the memory of the chip together with the algorithm used to generate it.

During the initialization, it may be provided that the public key and its hash value are input so that the check can be carried out by comparing the stored hash value and the hash value newly input during the initialization.

However, it is also possible and proposed by the invention that the hash value of the input public key is calculated using the algorithm and the result is simply compared with the stored value. This is also a possibility for checking the correctness of the input public key.

Another possibility for checking may consist in the fact that, during the initialization of the chip card, the public key and the algorithm used to generate its hash value are input.

If a chip card manufacturer has a number of purchasers, it may be provided according to the invention that a public key or its hash value and possibly the algorithm required to calculate it are stored in a chip card for each of the possible purchasers. During the initialization, identification then takes place in any known manner to identify the purchaser in question. However, the checking of the hash value is carried out in the manner described herein.

According to the invention, in order to further improve security, it may also be provided that a number of public keys or hash values for a number of keys are stored for a purchaser, for example in order in this way to use keys of different length.

Further features, details and advantages of the invention emerge from the following description of one preferred embodiment of the invention and with reference to the drawing, in which

Fig. 1 schematically shows the structure of the chip of a chip card;

Fig. 2 schematically shows the structure of the chip following the introduction of the initialization image;

Fig. 3 shows the introduction of the purchaser-specific secret key;

Fig. 4 shows the introduction of the test data at the chip card manufacturer;

Fig. 5 shows the status of the chip following a successful check.

The chip contains a ROM mask 1 which is produced by the ROM mask manufacturer and is introduced into the chip by the chip manufacturer. The ROM mask contains among other things the operating system which is required for the further manufacturing steps and for operation of the chip.

The chip furthermore contains an EEPROM 2 which is designed to receive data and program code. The EEPROM 2 is divided into three areas, namely a start area 3, a test area 4 and an area 5 for data and program code.

Fig. 1 shows the status at the chip manufacturer, in the secure environment 6 of which the key of the chip manufacturer is written from a memory area 7, in a secure manner, to a memory area 8 of the start area 3 of the EEPROM.

In detail, the following applies:

Key management of the chip manufacturer and of the chip manufacturer's purchaser, e.g. the distributor:

During chip production, the chip manufacturer introduces the ROM mask into the evaluated chip hardware. The

production environment of the chip manufacturer must be evaluated in accordance with the provisions of the Digital Signature Act (SigG) for the production of chips complying with SigG. The chip manufacturer confirms to the distributor and to the chip card manufacturer that only chips with evaluated hardware are used to produce signature card chips.

ROM mask of the chip:

The ROM mask manufacturer creates the operating system and application software for the chip card in the form of a ROM mask.

The ROM mask contains, for each purchaser, for example a distributor, two 20-byte-long hash values relating to a distributor-specific public key PK-distributor-chip and also the 2-byte-long byte lengths of the modulus and the 3-byte-long key identifier Info-PK-distributor for the respective PK-distributor-chip.

To this end, each distributor provides the ROM mask manufacturer beforehand with these values for the hash value, byte length and Info-PK-distributor. In addition, the ROM mask manufacturer also receives the modulus for calculating the hash value.

Fig. 1 shows the arrangement of the hash values in fields 9 to 12 of the ROM mask of the chip; each hash value is supplemented in the ROM mask by the respective additional information as described above.

The ROM mask is then evaluated by the evaluator in accordance with the security requirements laid down by the Digital Signature Act in respect of the technical component for generating and storing the signature key, and is handed over to the chip manufacturer so as to be applied to the evaluated chip.

During chip manufacture, the chip manufacturer introduces into the start area 3 of the EEPROM (usual size 64 bytes) of the chip card, in a secure manner, the triple-DES key K-chip together with additional information regarding the key, the chip password and further data.

The operating system of the chip must ensure, by means of suitable measures, that the incorporated key K-chip and the chip password cannot be read from the EEPROM and the start area cannot be manipulated. Furthermore, it must be possible for the chip password and the key K-chip to be introduced only into the start area of the EEPROM.

The operating system of the chip is to be configured in such a way that sub-program call-ups which are initiated by the ROM code in order to be able to address code in the EEPROM area, e.g. always point to one or more corresponding branch addresses 13 in the start area of the EEPROM. These branch addresses 13 always contain a "RETURN" instruction from the time of chip production at the chip manufacturer to successful execution of the VERIFY_EEPROM command, i.e. the other EEPROM areas are addressed neither directly nor indirectly to execute code.

Manufacture of the chip of the chip card at the chip manufacturer is thus complete. Following modularization, the chips produced are delivered to the chip card manufacturer. The chip card manufacturer receives chips for the chip card which are not yet purchaser-specific but rather are assigned to one of the four purchasers for example only during the initialization phase. This simplifies the availability of the chip quantities at the chip card manufacturer and reduces the unit price of the chip on account of the greater quantities purchased from the chip manufacturer.

Key exchange with the distributors in respect of K-chip

Each chip manufacturer introduces its K-chip into the security module (S-box) of the initialization tool, which is installed for this purpose at the respective purchaser/distributor. This S-box has inter alia the following functionality:
Introduction of the chip-manufacturer-specific key K-chip;
Introduction of the distributor-specific key K-chip-distributor;
Encryption of the introduced K-chip-distributor with the key K-chip;
Encryption of the key for the test area of the chip with K-chip-distributor;
Calculation of the signature via the test values and
Creation of the corresponding codes for the test cards.

The cryptogram relating to K-chip-distributor is subsequently transferred to the initialization table of the chip of the chip card. A clear delimitation of the security concept of the individual distributors is achieved by the distributor-specific K-chip-distributor.

Given any knowledge of a key K-chip-distributor of a different distributor which is encrypted with K-chip, it must not be possible for the distributors to use the key K-chip-distributor of a different distributor. Since K-chip is chip-manufacturer-specific, the function "Export of the K-chip" and "Decryption with K-chip" must be blocked in the S-box of the initialization tool. It must only be possible to encrypt K-chip-distributor by means of this S-box. It should be possible for the functions of the S-box to be implemented only after previous authentication (e.g. PIN input) of the user with the S-box.

Initialization and construction of the initialization table:

The initialization table is an essential production means for the chip card. It can be used during manufacture of the chip card to introduce identical memory contents into all the chip cards of a ROM mask.

After the programming of code and data structures is complete, the ROM mask manufacturer creates an image of a specific state of the persistent memory of the chip, the so-called initialization image. This is to be loaded into the persistent memory of the chip card chip in order to prepare it to receive personalization data. For the purpose of appropriate and reliable introduction, the image for the chip must be transformed, by adding control, protocol and test information, into the format of a productive initialization table for the initialization system.

The initialization table contains, after the table header, sub-tables with command sequences which the initializer must transfer to the chip card and also commands and information for controlling and protocolling the initialization process.

The parts of the initialization table which contain the commands and the associated command data are loaded by the initializer into the chip card in sets, by sending the corresponding data set, consisting of a basic command and the associated data, to the chip. The chip responds with a return code which has to be compared with the corresponding return code in the command table. If the two do not match, the differing return code must be protocolled and the loading of the initialization table must be stopped.

Fig. 2 schematically shows the structure of the EEPROM of the chip card following introduction of the initialization image. The area 5 of the EEPROM which is provided for the program code and data now contains a batch table 15, from which branches may issue to a number of memory areas 16 with program code. The area 5 also contains an area 17 for a file system with constant data contents.

Introduction of the distributor-specific K-chip-distributor at the chip card manufacturer:

As the first action of the initialization operation at the chip card manufacturer, the distributor-specific key K-chip-distributor is introduced. The cryptogram of the key has been securely provided by the distributor to the chip card manufacturer beforehand as part of the initialization table.

The initialization system sends a command VERIFY_CHIPPWD with the encrypted K-chip-distributor to the chip of the signature card. Following receipt of the data, the chip compares the transmitted key information items (VID, KID and KV) with the values stored in the start area. If the chip detects that a new key has been transmitted to it, the chip decrypts the cryptogram of K-chip-distributor with the aid of the key K-chip. In the start area of the EEPROM, the key K-chip is replaced by K-chip-distributor. Successful execution of the command is protocolled by changing the chip status. Fig. 3 shows the introduction of the key K-chip-distributor. Like in the drawing shown in Fig. 1, the secret key of the purchaser is transferred from a memory space 18 of the chip card manufacturer to the memory area 8, in which the secret key of the chip manufacturer has been accommodated until now.

Description of the command "VERIFY_CHIPPWD":

The command VERIFY_CHIPPWD makes it possible either to replace the key K_{chip} with the distributor-specific key $K_{chip\text{-}distributor}$ (if L_c = '1E') or to verify the password transmitted in the command data by comparing it with the chip password stored in the persistent memory (if L_c = '08' or L_c = '1E'). In each case, successful execution of the command authorizes the outside world to execute further commands.

The error counter (EC) for the chip password and the error counter for $K_{chip\text{-}distributor}$ must be persistently stored in the start area of the EEPROM in order that they are not erased in the event of a power failure. If, upon use of the chip password or of $K_{chip\text{-}distributor}$ by the command, the respective EC has the value '00', the command is terminated and issues an error report.

When the command is called up, firstly the integrity of the start area is checked by means of a routine to be implemented in a proprietary manner.

For the "Compare chip password" mode (L_c = '08'), the procedure is as follows:

If the value of the error counter for the chip password is '00', the command is terminated with the return code '69 83'. If this EC is not '00', the chip verifies the 8-byte-long chip password CHIPPWD transmitted in the command data by comparing it with the chip password which exists in the start area of the EEPROM. In the event of an incorrect value of the chip password, the EC of the chip password is decremented by one and the command is terminated with the return code '63 Gx'. Here, 'x' indicates the value of this EC and thus the number of further attempts, that is to say 'x' = '2', '1' or '0'.

If the comparison is successful, the error counter of the chip password is reset to the initial value '03', a flag

which indicates successful verification of the chip password is placed in the volatile memory, and the command is terminated with the outputting of the return code '90 00'.

The command call-up in the mode L_c = '08' is used for example as authentication during the introduction of the protocol data of module manufacture, during personalization and - in the case of an initialization table which is split into two - at the start of the second part of the initialization table. At the start of the initialization table, the mode L_c = '1E' is used since a key change is provided there.

In the text which follows, the following references will be used:

Vdata: VID2 || KID2 || KV2 || $K_{chip-distributor}$ || '00 00 00 00 00'
MAC(Vdata): retail-CFB-MAC relating to Vdata calculated with $K_{chip-distributor}$ and ICV = '00...00'
CHIPPWD: an 8-byte-long password or MAC(Vdata) provided by the ROM mask manufacturer and introduced by the chip manufacturer, if a key change to $K_{chip-distributor}$ has already taken place
[$K_{chip-distributor}$]: $K_{chip-distributor}$ encrypted with K_{chip} in the CBC mode with ICV = '00...00' triple-DES
VID1: ZKA manufacturer identifier of the chip manufacturer for the K_{chip} contained in the chip
VID2: ZKA manufacturer identifier of the distributor for the $K_{chip-distributor}$ to be introduced into the chip
KID1, KID2: key number/ID of the corresponding key
KV1, KV2: key version of the corresponding key

For the mode "Possibly change key" (L_c = '1E'), the procedure is as follows:

The chip checks whether the triple (VID1, KID1, KV1) is different from the triple (VID2, KID2, KV2) and whether the triple (VID, KID, KV) which is located in the start area and belongs to K_{chip} is the same as either (VID1, KID1, KV1) or (VID2, KID2, KV2). If this is not the case, the command is terminated with the return code '64 00'.

L_c = '1E' with key change:

If the triple (VID, KID, KV) is identical to (VID1, KID1, KV1), the value of the error counter for the key K_{chip} is checked. If it is '00', the command is terminated with the return code '69 83'. If this EC is not '00', the chip checks the command data. The cryptogram [$K_{chip\text{-}distributor}$] is decrypted with K_{chip}. Using the $K_{chip\text{-}distributor}$ thus obtained, the value MAC(Vdata) is then calculated and compared with the corresponding value from the command data. If the two MAC values do not match, the EC of K_{chip} is decremented by one and the command is terminated with the return code '63 Cx'. Here, 'x' indicates the value of this error counter and thus the number of further attempts, that is to say 'x' = 'F'...'0'.

If the MAC from the command data matches the calculated MAC(Vdata), in the start area the chip replaces VID, KID and KV of the key K_{chip} by VID2, KID2 and KV2 and K_{chip} by $K_{chip\text{-}distributor}$ and sets the associated EC to '10'. The 8-byte-long chip password in the start area is then replaced by the value MAC(Vdata) (=CHIPPWD) and the EC of the chip password is set to '03'. The chip status is set to 2. A flag which indicates successful verification of the chip password is then placed in the volatile memory, and the command is terminated with the return code '90 00'.

L_c = '1E' without key change:

If the triple (VID, KID, KV) is identical to (VID2, KID2, KV2), CHIPPWD in the start area is compared with the value MAC(Vdata) transmitted in the command data. This check proceeds in the same way as in the mode L_c = '08', with the exception of the change in chip status.

If the value of the error counter of the chip password is '00', the command is terminated with the return code '69 83'. If the two values for CHIPPWD do not match, the EC of the chip password is decremented by one and the command is terminated with the return code '63 Cx'. Here, 'x' indicates the value of this EC and thus the number of further attempts, that is to say 'x' = '2', '1' or '0'. If the comparison is successful, the EC of the chip password is reset to the initial value '03' and a flag which indicates successful verification of the chip password is placed in the volatile memory. The chip status is set to 2 and the command is terminated with the outputting of the return code '90 00'.

Apart from the change from K_{chip} to $K_{chip\text{-}distributor}$, it is also possible by means of this command in the mode L_c = '1E' to change from one distributor key $K_{chip\text{-}distributor}$ to a different distributor key $K_{chip\text{-}distributor}$. In this case, the old $K_{chip\text{-}distributor}$ in the chip is treated in the manner described above for K_{chip}. This requires the corresponding command data with the code of the new $K_{chip\text{-}distributor}$ below the old one and the new chip password.

Function of the command:

Checking and processing the chip password, possible key exchange, input length: L_c '1E' or '08'

Command data:
if L_c = '1E': VID1 || KID1 || KV1 || VID2 || KID2 || KV2 || [$K_{chip\text{-}distributor}$] || MAC(Vdata)
if L_c = '08': 8-byte-long chip password CHIPPWD

Return codes:
'90 00': successfully executed
'6E 00': invalid CLA value
'6D 00': invalid INS value
'6A 86': invalid value in P1 or P2
'67 00': incorrect length
'6F 00': general error - technical problem
'63 Cx': authentication failed, 'x' further attempts possible
'69 83': authentication blocked (error counter = '00')
'64 00': execution error, state of non-volatile memory unchanged (output when content errors or inconsistencies are detected during the reading of data)

The rest of the initialization process for the chip of the signature card is effected by loading the initialization table into the chip. In order to satisfy the high security requirements placed on a signature card, the initialization image is protected against any manipulation.

Loading of the initialization image into the chip:

In the initialization environment of the chip card manufacturer, following introduction of the distributor-specific key K-chip-distributor into the memory space 21, firstly the area boundaries BTAB for the initialization image are transferred. To this end, the part CTRL_TAB of the initialization table contains the INITIALIZE command with the parameter BTAB. The chip accepts the values defined in BTAB for the area boundaries only if the area is restricted or remains the same compared to the provisions of BChip in the start area of the chip. The chip status then changes to "Load image". Subsequent overwriting of BTAB is ruled out by the chip until the VERIFY_EEPROM command has been successfully completed or until reinitialization. It must in no way be possible to

specify and manipulate parts of the start area of the EEPROM or other memory areas outside the code/data area as the address area to be initialized.

In the next step, the code/data area of the EEPROM is initialized by the initialization system of the chip card manufacturer on the basis of the initialization table as shown in Fig. 2, wherein here the address areas of the EEPROM which are to be initialized are checked by the operating system with respect to their position within the area boundaries defined by BTAB.

The data are then introduced into the test area, cf. Fig. 4. Besides the aforementioned memory space 21 for the distributor-specific key K-chip-distributor, there are other memory spaces 20 and 22 to 24 which are occupied by test data during the initialization.

The test data contain (on the interface between initialization machine and chip card):
the triple-DES key KINITAB_MAC encrypted with K-chip-distributor,
the triple-DES personalization key KPers encrypted with K-chip-distributor,
the triple-DES personalization key KTransfer encrypted with K-chip-distributor,
the correspondence Z with the semiconductor/ROM mask combination which is checked during VERIFY_EEPROM against an existing entry in the start area of the chip EEPROM,
the MAC relating to the code/data area,
the additional information item Info-PK-distributor relating to PK-distributor-chip,
the public key PK-distributor-chip,
the distributor-specific identifier of the initialization, and
the signature relating to the hash value of KINITAB_MAC || KPers || KTransfer || Z || BTAB || MAC (image) ||

Info-PK-distributor || distributor-specific identifier of the initialization.

Thereafter, a check takes place as to whether the contents of the code/data area 5 of the EEPROM 2 of the chip card are authentic. To this end, a corresponding command is sent to the chip. This leads to the following processes: The operating system of the chip firstly forms the hash value relating to the public key PK-distributor-chip stored in the test area of the memory, and compares it with the hash value referenced for Info-PK-distributor which is stored in the ROM mask. If the calculated hash value matches the associated hash value which exists in the ROM mask, the PK-distributor-chip stored in the test area is authentic.

The chip then checks, using the key PK-distributor-chip, the signature relating to the test data P = (KINITAB_MAC || KPers || KTransfer || Z || BTAB || MAC relating to the initialization image || Info-PK-distributor || distributor-specific identifier of the initialization). To this end, firstly the hash value (SHA-1) relating to P is formed and then it is compared with the hash value which results from RSA public key encryption of the signature relating to P with PK-distributor-chip. If the two hash values match, in particular the introduced KINITAB_MAC and the MAC relating to the EEPROM contents of the code/data area are authentic. The test data are accepted only if Z matches the corresponding identifier of the chip manufacturer data in the start area.

The chip then calculates, with the key KINITAB_MAC using the area boundaries BTAB, the MAC relating to the code/data area of the EEPROM (including the protocol data for chip manufacture (bytes 1-3, 8-9 and 14), for initialization (without the first 16 bytes) and for personalization) and compares it with the MAC stored in the test area. If the two MACs match, it is detected that

the EEPROM has been correctly initialized and the introduced initialization image is authentic.

After a successful check, the chip changes its status to OK. The operating system can then replace the RETURN instruction stored in the batch address 13 with the address of the batch table 15 in the code/data area 5 of the EEPROM 2. The sub-program call-ups of the ROM code are thus no longer blocked but rather are addressed via the batch table or another mechanism on the corresponding program code in the corresponding area 5 of the EEPROM 2. The program code in this area is thus available for the operating system and can be executed. This is shown schematically in Fig. 5.

The personalization can then be carried out. The organizational separation of initialization environment and personalization environment, which has up to now been strict, need not be retained during production of the chip card since the keys are incorporated in the chip card in an encrypted manner.